Polar Stereographic map projection computation of ϕ_1 from k_0

C. Rollins, NGA-SNAC, February 27, 2009

This work is undertaken pursuant to the responsibility of Coordinate Systems Analysis, branch SNAC of NGA, to support the development of GEOTRANS.

Introduction

The explanations to follow will refer to the Stereographic Projection of the North polar regions. Readers will realize that the situation with the South Polar Stereographic projection is symmetric.

The Polar Stereographic map projection, as defined by ISO/IEC 18026:2006 and other places, allows a factor to adjust the scale of the map projection. This number is called the "central scale" in the above ISO document, and is notated k_0 there. When it is 1, the local scale of the map projection at the North Pole is 1. It has to be between 0 (exluded) and 1 (included). That is, $0 < k_0 \le 1$. The UPS grid system employs the Polar Stereographic projection using $k_0 = 0.994$.

If $k_0 < 1$, then there is a parallel circle at latitude ϕ_1, along which the local scale is exactly 1. The number ϕ_1 is called the "latitude of unity scale". Also, the parallel circle at this latitude is called the "standard parallel". Any arc of the standard parallel defined by an interval of longitude $\lambda_2 - \lambda_1$ has the same length on the map projection plane as it does on the ellipsoid. The formula is

$$\text{arclength} = a\,(\lambda_2 - \lambda_1) * \frac{\mathrm{Cos}[\phi_1]}{\sqrt{1 - \epsilon^2\,\mathrm{Sin}[\phi_1]^2}}$$

where "a" is the semi-major axis of the ellipsoid, and ϵ is the eccentricity of the ellipsoidal, and λ_1 and λ_2 are the longitudes (in radians) that mark the endpoints of the segment of the parallel circle. The reader will understand how to take into account the discontinuity of longitude that occurs as specified, often at $\lambda = \pm\pi = \pm 180\,°$.

Generally speaking, the Polar Stereographic projection enlarges or shrinks the parallel circles. Only the standard parallel is portrayed arclength-true.

k_0 as a function of ϕ_1

The two quantities, k_0 and ϕ_1 are dependent. k_0 is straightforwardly a function of ϕ_1 as follows:

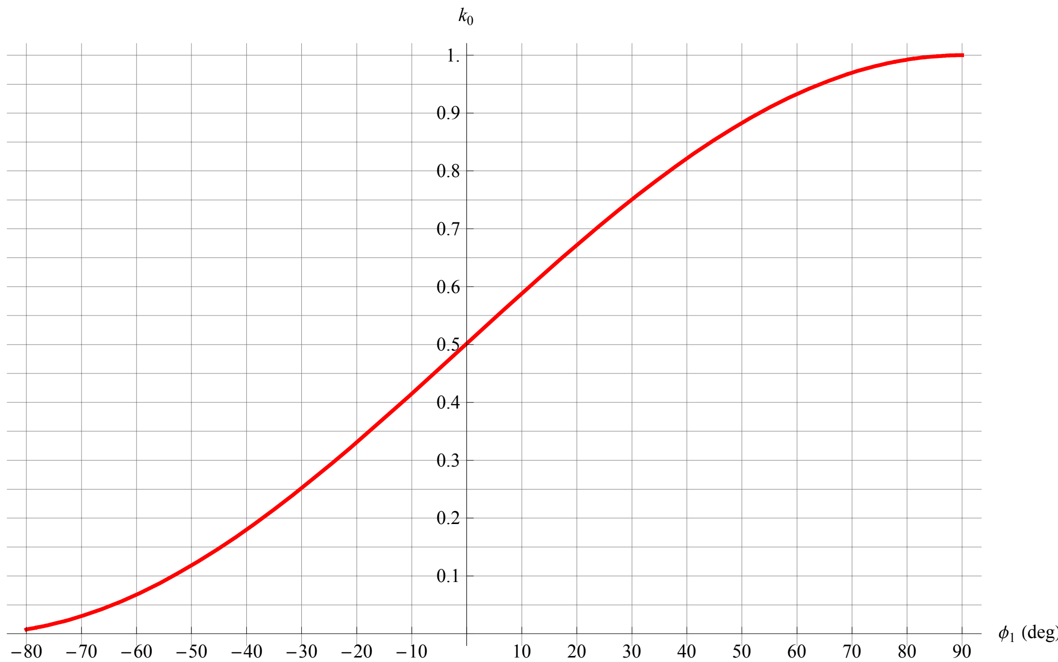
$$k_0 = \frac{1 + \mathrm{Sin}[\phi_1]}{2} * \frac{k_{90}}{\sqrt{(1 + \epsilon\,\mathrm{Sin}[\phi_1])^{1+\epsilon}\,(1 - \epsilon\,\mathrm{Sin}[\phi_1])^{1-\epsilon}}} \qquad (1)$$

where k_{90} is defined

$$k_{90} = \sqrt{(1 + \epsilon)^{1+\epsilon}\,(1 - \epsilon)^{1-\epsilon}}$$

Substitute $\phi_1 = 90\,°$ into Equation (1) to see that $k_0 = 1$ in that case. Equation (1) may be derived from the discussion of Polar Stereographic projection in Table 5.22 of ISO/IEC 18026:2006.

Observe the following plot of k_0 versus ϕ_1 for the WGS84 ellipsoid

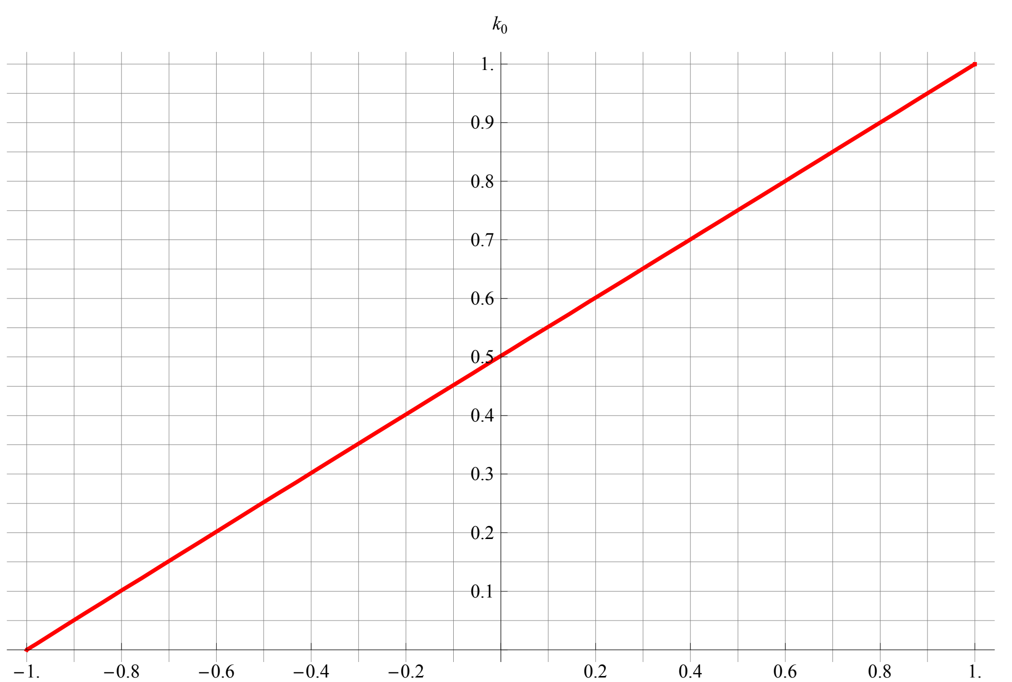


Despite appearances, if the Equator is the standard parallel, k_0 is not exactly 0.5000. It is $k_{90}/2$ or 0.501678 in the case of the WGS84 ellipsoid. See Equation (1).

■ k_0 as a function of $s = \mathrm{Sin}[\phi_1]$

Look again at the above plot of k_0 versus ϕ_1. Just where the graph holds most interest, i.e. near the Pole, i.e. near $\phi_1 \simeq 90\,°$ is where the curve is flat, making numerical difficulties for the problem of computing ϕ_1, given k_0.

It is informative to see how the bend in the curve is caused by the sine function. When we take that out, by plotting k_0 versus $s = \mathrm{Sin}[\phi_1]$ instead, we get:



One should flatten the ellipsoid severely, such as flattening $= 1/20$ in order to see the important departures from the simple curves these appear to be.

ϕ_1 as a function of k_0

The more difficult conversion is the inverse relationship — to compute ϕ_1, given k_0, Although Equation (1) is not easily solved for ϕ_1, the following equation obviously holds: (For convenience, ϕ_1 is shortened to ϕ).

$$\mathrm{Sin}[\phi] = \frac{2\, k_0 \sqrt{(1 + \epsilon\, \mathrm{Sin}[\phi])^{1+\epsilon}\, (1 - \epsilon\, \mathrm{Sin}[\phi])^{1-\epsilon}}}{k_{90}} - 1 \tag{2}$$

Equation (2) suggests the substitution $s = \mathrm{Sin}[\phi]$, and the following iteration scheme:

$$s_{k+1} = \frac{2\, k_0 \sqrt{(1 + \epsilon\, s_k)^{1+\epsilon}\, (1 - \epsilon\, s_k)^{1-\epsilon}}}{k_{90}} - 1 \tag{3}$$

where s_k is the k^{th} iterate of trying to find $s = \mathrm{Sin}[\phi]$ that satisfies Equation (2).

If a good starting value can be found, or even if a lousy starting value is used, but there is always convergence, then this is an algorithm for the desired function $k_0 \longrightarrow \phi_1$. The starting value $s_0 = 0$ works. So also does $s_0 = -1 + 2\, k_0$ which is slightly more efficient, generally. Other starting values can be found by linear interpolation of the k_0 versus "s" plot, or linear interpolation of the portion of the plot of interest.

■ Other iteration algorithms

A cheap version of Newton-Raphson iteration was tried using

$$s_{k+1} = s_k + 2\, (k_0 - f[s_k]) \tag{4}$$

where $f[s]$ = the right hand side of Equation (1) evaluated at $\mathrm{Sin}[\phi_1] = s$. The cheapness is in the approximation of the derivative as $f'[s] \simeq \frac{1}{2}$.

It works, but not better.